Named Entity Recognition with Small Strongly Labeled and Large Weakly Labeled Data

Haoming Jiang*[1], Danqing Zhang[2], Tianyu Cao[2], Bing Yin[2], Tuo Zhao[1]
[1]Georgia Institute of Technology, Atlanta, GA, USA
[2]Amazon.com Inc, Palo Alto, CA, USA
{jianghm, tourzhao}@gatech.edu
{danqinz, caoty, alexbyin}@amazon.com

Abstract

Weak supervision has shown promising results in many natural language processing tasks, such as Named Entity Recognition (NER). Existing work mainly focuses on learning deep NER models only with weak supervision, i.e., without any human annotation, and shows that by merely using weakly labeled data, one can achieve good performance, though still underperforms fully supervised NER with manually/strongly labeled data. In this paper, we consider a more practical scenario, where we have both a small amount of strongly labeled data and a large amount of weakly labeled data. Unfortunately, we observe that weakly labeled data does not necessarily improve, or even deteriorate the model performance (due to the extensive noise in the weak labels) when we train deep NER models over a simple or weighted combination of the strongly labeled and weakly labeled data. To address this issue, we propose a new multi-stage computational framework – NEEDLE with three essential ingredients: (1) weak label completion, (2) noise-aware loss function, and (3) final fine-tuning over the strongly labeled data. Through experiments on E-commerce query NER and Biomedical NER, we demonstrate that NEEDLE can effectively suppress the noise of the weak labels and outperforms existing methods. In particular, we achieve new SOTA F1-scores on 3 Biomedical NER datasets: BC5CDR-chem 93.74, BC5CDR-disease 90.69, NCBI-disease 92.28.

1 Introduction

Named Entity Recognition (NER) is the task of detecting mentions of real-world entities from text and classifying them into predefined types. For example, the task of E-commerce query NER is to identify the product types, brands, product attributes of a given query. Traditional deep learning approaches mainly train the model from scratch (Ma and Hovy, 2016; Huang et al., 2015), and rely on large amounts of labeled training data. As NER tasks require token-level labels, annotating a large number of documents can be expensive, time-consuming, and prone to human errors. Therefore, the labeled NER data is often limited in many domains (Leaman and Gonzalez, 2008). This has become one of the biggest bottlenecks that prevent deep learning models from being adopted in domain-specific NER tasks.

To achieve better performance with limited labeled data, researchers resort to large unlabeled data. For example, Devlin et al. (2019) propose to pre-train the model using masked language modeling on large unlabeled open-domain data, which is usually *hundreds/thousands of times larger* than the manually/strongly labeled data. However, open-domain pre-trained models can only provide limited semantic and syntax information for domain-specific tasks. To further capture domain-specific information, Lee et al. (2020); Gururangan et al. (2020) propose to continually pre-train the model on large in-domain unlabeled data.

When there is no labeled data, one approach is to use weak supervision to generate labels automatically from domain knowledge bases (Shang et al., 2018; Liang et al., 2020). For example, Shang et al. (2018) match spans of unlabeled Biomedical documents to a Biomedical dictionary to generate weakly labeled data. Shang et al. (2018) further show that by merely using weakly labeled data, one can achieve good performance in biomedical NER tasks, though still underperforms supervised NER models with manually labeled data. Throughout the rest of the paper, we refer to the manually labeled data as strongly labeled data for notational convenience.

While in practice, we often can access both a small amount of strongly labeled data and a large

* Work was done during internship at Amazon.

amount of weakly labeled data, generated from large scale unlabeled data and domain knowledge bases. A natural question arises here:

"Can we simultaneously leverage small strongly and large weakly labeled data to improve the model performance?"

The answer is yes, but the prerequisite is that you can properly suppress the extensive labeling noise in the weak labels. The weak labels have three features: 1) "incompleteness": some entity mentions may not be assigned with weak labels due to the limited coverage of the knowledge base; 2) "labeling bias": some entity mentions may not be labeled with the correct types, and thus weak labels are often noisy; 3) "ultra-large scale": the weakly labeled data can be *hundreds/thousands of times larger* than the strongly labeled data.

An ultra-large volume of weakly labeled data contains useful domain knowledge. But it also comes with enormous noise due to the "incompleteness" and "labeling bias" of weak labels. The enormous noise can dominate the signal in the strongly and weakly labeled data, especially when combined with the unsupervised pre-training techniques. Such noise can be easily overfitted by the huge neural language models, and may even deteriorate the model performance. This is further corroborated by our empirical observation (See Section 4) that when we train deep NER models over a simple or weighted combination of the strongly labeled and weakly labeled data, the model performance almost always becomes worse.

To address such an issue, we propose a three-stage computational framework named NEEDLE (**N**oise-aware w**E**akly sup**E**rvise**D** continua**L** pr**E**-training). At Stage I, we adapt an open-domain pre-trained language model to the target domain by in-domain continual pre-training on the large in-domain unlabeled data. At Stage II, we use the knowledge bases to convert the in-domain unlabeled data to the weakly labeled data. We then conduct another continual pre-training over both the weakly and strongly labeled data, in conjunction with our proposed weak label completion procedure and noise-aware loss functions, which can effectively handle the"incompleteness" and "noisy labeling" of the weak labels. At Stage III, we fine-tune the model on the strongly labeled data again. The last fine-tuning stage is essential to the model fitting to the strongly labeled data.

We summarize our key contributions as follows:

- We identify an important research question on weak supervision: while training deep NER models using a simple or weighted combination of the strongly labeled and weakly labeled data, the ultra-large scale of the weakly labeled data aggravates the extensive noise in the weakly labeled data and can significantly deteriorate the model performance.
- We propose a three-stage computational framework named NEEDLE to better harness the ultra-large weakly labeled data's power. Our experimental results show that NEEDLE significantly improves the model performance on the E-commerce query NER tasks and Biomedical NER tasks. In particular, we achieve new SOTA F1-scores on 3 Biomedical NER datasets: BC5CDR-chem 93.74, BC5CDR-disease 90.69, NCBI-disease 92.28. We also extend the proposed framework to the multi-lingual setting.

2 Preliminaries

We briefly introduce the NER problem and the unsupervised language model pre-training.

2.1 Named Entity Recognition

NER is the process of locating and classifying named entities in text into predefined entity categories, such as products, brands, diseases, chemicals. Formally, given a sentence with N tokens $X = [x_1, ..., x_N]$, an entity is a span of tokens $s = [x_i, ..., x_j]$ $(0 \leq i \leq j \leq N)$ associated with an entity type. Based on the `BIO` schema (Li et al., 2012), NER is typically formulated as a sequence labeling task of assigning a sequence of labels $Y = [y_1, ..., y_N]$ to the sentence X. Specifically, the first token of an entity mention with type `X` is labeled as `B-X`; the other tokens inside that entity mention are labeled as `I-X`; and the non-entity tokens are labeled as `O`.

Supervised NER. We are given M sentences that are already annotated at token level, denoted as $\{(X_m, Y_m)\}_{m=1}^M$. Let $f(X; \theta)$ denote an NER model, which can compute the probability for predicting the entity labels of any new sentence X, where θ is the parameter of the NER model. We train such a model by minimizing the following loss over $\{(X_m, Y_m)\}_{m=1}^M$:

$$\hat{\theta} = \operatorname{argmin}_{\theta} \frac{1}{M} \sum_{m=1}^{M} \ell(Y_m, f(X_m; \theta)), \quad (1)$$

where $\ell(\cdot,\cdot)$ is the cross-entropy loss for token-wise classification model or negative likelihood for CRF model (Lafferty et al., 2001).

Weakly Supervised NER. Previous studies (Shang et al., 2018; Liang et al., 2020) of weakly supervised NER consider the setting that no strong label is available for training, but only *weak labels* generated by matching unlabeled sentences with external gazetteers or knowledge bases. The matching can be achieved by string matching (Giannakopoulos et al., 2017), regular expressions (Fries et al., 2017) or heuristic rules (e.g., POS tag constraints). Accordingly, they learn an NER model by minimizing Eq. (1) with $\{Y_m\}_{m=1}^{M}$ replaced by their weakly labeled counterparts.

2.2 Unsupervised Pre-training

One of the most popular approaches to leverage large unlabeled data is unsupervised pre-training via masked language modeling. Pre-trained language models, such as BERT and its variants (e.g., RoBERTa Liu et al. (2019), ALBERT Lan et al. (2020b) and T5 Raffel et al. (2019)), have achieved state-of-the-art performance in many natural language understanding tasks. These models are essentially massive neural networks based on bi-directional transformer architectures, and are trained using a tremendous amount of open-domain data. For example, the popular BERT-base model contains 110 million parameters, and is trained using the BooksCorpus (Zhu et al., 2015) (800 million words) and English Wikipedia (2500 million words). However, these open-domain data can only provide limited semantic and syntax information for domain-specific tasks. To further capture domain-specific knowledge, Lee et al. (2020); Gururangan et al. (2020) propose to continually pre-train the model over large in-domain unlabeled data.

3 Method

To harness the power of weakly labeled data, we propose a new framework — NEEDLE, which contain stages as illustrated in Figure 1:

1) We first adapt an open-domain pre-trained language model to the downstream domain via MLM continual pre-training on the unlabeled in-domain data.

2) We use the knowledge bases to convert the unlabeled data to the weakly labeled data through weak supervision. Then we apply noise-aware continual pre-training for learning task-specific knowledge from both strongly and weakly labeled data;

3) Lastly, we fine-tune the model on the strongly labeled data again.

3.1 Stage I: Domain Continual Pre-training over Unlabeled Data

Following previous work on domain-specific BERT (Gururangan et al., 2020; Lee et al., 2020), we first conduct domain continual masked language model pre-training on the large in-domain unlabeled data $\{\tilde{X}_m\}_{m=1}^{\tilde{M}}$. Note that the masked language model $f_{\rm LM}(\cdot;\theta_{\rm enc},\theta_{\rm LM})$ contains encoder parameters $\theta_{\rm enc}$ and classification head parameters $\theta_{\rm LM}$, which are initialized from open-domain pre-trained masked language models (e.g., BERT and RoBERTa).

3.2 Stage II: Noise-Aware Continual Pre-training over both Strongly and Weakly labeled Data

In the second stage, we use the knowledge bases to convert the unlabeled data to weakly labeled data to generate weak labels for the unlabeled data: $\{(\tilde{X}_m,\tilde{Y}_m^w)\}_{m=1}^{\tilde{M}}$. We then continually pre-train the model with both weakly labeled in-domain data and strongly labeled data. Specifically, we first replace the MLM head by a CRF classification head (Lafferty et al., 2001) and conduct noise-aware weakly supervised learning, which contains two ingredients: *weak label completion procedure* and *noise-aware loss function*.

• **Weak Label Completion.** As the weakly labeled data suffer from severe missing entity issue, we propose a weak label completion procedure. Specifically, we first train an initial NER model $f(;\theta^{\rm Init})$ by optimizing Eq (1) with $\theta^{\rm Init}=(\theta_{\rm enc},\theta_{\rm CRF})$, where the encoder $\theta_{\rm enc}$ is initialized from Stage I and NER CRF head $\theta_{\rm CRF}$ is randomly initialized. Then, for a given sentence $\tilde{X}=[x_1,...,x_N]$ with the original weak labels $\tilde{Y}^w=[y_1^w,...,y_N^w]$ and the predictions from the initial model $\tilde{Y}^p=\mathrm{argmin}_Y\,\ell(Y,f(\tilde{X};\theta^{\rm Init}))=[y_1^w,...,y_N^w]$, we generate the corrected weak labels $\tilde{Y}^c=[y_1^c,...,y_N^c]$ by:

$$y_i^c=\begin{cases} y_i^p & \text{if } y_i^w=\text{O (non-entity)} \\ y_i^w & \text{otherwise} \end{cases} \tag{2}$$

Such a weak label completion procedure can remedy the incompleteness of weak labels.

Figure 1: Three-stage NEEDLE Framework.

Noise-Aware Loss Function. The model tends to overfit the noise of weak labels when using negative log-likelihood loss over the weakly labeled data, Eq.(1). To alleviate this issue, we propose a noise-aware loss function based on the estimated confidence of the corrected weak labels $\tilde{Y}^c$, which is defined as the estimated probability of $\tilde{Y}^c$ being the true labels $\tilde{Y}$: $\hat{P}(\tilde{Y}^c = \tilde{Y}|\tilde{X})$. The confidence can be estimated by the model prediction score $f(\tilde{X};\theta)$ and histogram binning (Zadrozny and Elkan, 2001). See more details in Appendix A.

We design the noise-aware loss function to make the fitting to the weak labels more conservative/aggressive, when the confidence is lower/higher. Specifically, when $\tilde{Y}^c = \tilde{Y}$, we let loss function $\mathcal{L}$ be the negative log-likelihood, i.e., $\mathcal{L}(\cdot,\cdot|\tilde{Y}^c = \tilde{Y}) = \ell(\cdot,\cdot)$; when $\tilde{Y}^c \neq \tilde{Y}$, we let $\mathcal{L}$ be the negative log-unlikelihood, i.e., $\mathcal{L}(\cdot,\cdot|\tilde{Y}^c \neq \tilde{Y}) = \ell^-(\cdot,\cdot)$ [1]. Accordingly, the noise-aware loss function is designed as

$$
\begin{aligned}
&\ell_{NA}(\tilde{Y}^c, f(\tilde{X};\theta)) \\
&= \mathbb{E}_{\tilde{Y}_m = \tilde{Y}^c_m|\tilde{X}_m} \mathcal{L}(\tilde{Y}^c_m, f(\tilde{X}_m;\theta), \mathbb{1}(\tilde{Y}_m = \tilde{Y}^c_m)) \\
&= \hat{P}(\tilde{Y}^c = \tilde{Y}|\tilde{X})\ell(\tilde{Y}^c, f(\tilde{X};\theta)) + \\
&\quad \hat{P}(\tilde{Y}^c \neq \tilde{Y}|\tilde{X})\ell^-(\tilde{Y}^c, f(\tilde{X};\theta)),
\end{aligned} \tag{3}
$$

where the log-unlikelihood loss can be viewed as regularization and the confidence of weak labels can be viewed as an adaptive weight. The training objective on both the strongly labeled data and weakly labeled data is:

$$
\begin{aligned}
\min_\theta \frac{1}{M+\tilde{M}} \Big[& \sum_{m=1}^{M} \ell(Y_m, f(X_m;\theta)) \\
+ & \sum_{m=1}^{\tilde{M}} \ell_{NA}(\tilde{Y}^c_m, f(\tilde{X}_m;\theta)) \Big],
\end{aligned} \tag{4}
$$

3.3 Stage III: Final Fine-tuning

Stages I and II of our proposed framework mainly focus on preventing the model from the overfitting to the noise of weak labels. Meanwhile, they also suppress the model fitting to the strongly labeled data. To address this issue, we propose to fine-tune the model on the strongly labeled data again. Our experiments show that such additional fine-tuning is essential.

4 Experiments

We use transformer-based open-domain pretrained models, e.g., BERT, mBERT, RoBERTa-Large, (Devlin et al., 2019; Liu et al., 2019) with a CRF layer as our base NER models. Throughout the experiments, we use the BIO tagging scheme (Carpenter, 2009). For Stages I and II, we train the models for one epoch with batch size 144. For Stage III, we use the grid search to find optimal hyperparameters: We search the number of epochs in $[1, 2, 3, 4, 5, 10, 15, 20, 25, 30, 50]$ and batch size in $[64, 144, 192]$. We use ADAM optimizer with a learning rate of 5×10^{-5} on the E-commerce query NER dataset. In the Biomedical NER experiments, we search the optimal learning rate in $[1 \times 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}]$. All implementations are based on transformers (Wolf et al., 2019). We use an Amazon EC2 virtual machine with 8 NVIDIA V100 GPUs.

---

[1] $\ell(Y, f(X;\theta)) = -\log P_{f(X;\theta)}(Y)$
$\ell^-(Y, f(X;\theta)) = -\log[1 - P_{f(X;\theta)}(Y)]$

Dataset	Number of Samples				Weak Label	
	Train	Dev	Test	Weak	Precision	Recall
E-commerce Query Domain						
En	187K	23K	23K	22M	84.62	49.52
E-commerce Query Domain (Multilingual)						
Mul-En	257K	14K	14K	17M	84.62	49.52
Mul-Fr	79K	4K	4K			
Mul-It	52K	3K	3K			
Mul-De	99K	5K	5K			
Mul-Es	64K	4K	4K			
Biomedical Domain						
BC5CDR Chem	5K	5K	5K	11M	92.08	77.40
BC5CDR Disease	5K	5K	5K	15M	94.46	81.34
NCBI Disease	5K	1K	1K			

Table 1: Data Statistics

4.1 Datasets

We evaluate the proposed framework on two different domains: E-commerce query domain and Biomedical domain. The data statistics are summarized in Table 1.

For E-commerce query NER, we consider two settings: english queries and multilingual queries. For English NER, there are 10 different entity types, while the multilingual NER has 12 different types. The queries are collected from search queries to a shopping website. The unlabeled in-domain data and the weak annotation is obtained by aggregating user behavior data collected from the shopping website. We give more details about the weakly labeled data in Appendix E.

For Biomedical NER, we use three popular benchmark datasets: BC5CDR-Chem, BC5CDR-Disease (Wei et al., 2015), and NCBI-Disease (Dogan et al., 2014). These datasets only contain a single entity type. We use the pre-processed data in BIO format from Crichton et al. (2017) following BioBERT (Lee et al., 2020) and PubMedBERT (Gu et al., 2020). We collect unlabeled data from PubMed 2019 baseline[2], and use the dictionary lookup and exact string match to generate weak labels[3]. We only include sentences with at least one weak entity label.

[2]Titles and abstract of Biomedical articles: https://ftp.ncbi.nlm.nih.gov/pubmed/baseline/

[3]We collect a dictionary containing 3016 chemical entities and 5827 disease entities.

Weak Labels Performance Table 1 also presents the precision and recall of weak labels performance on a evaluation golden set. As can be seen, the weak labels suffer from severe incompleteness issue. In particular, the recall of E-commerce query NER is lower than 50. On the other hand, the weak labels also suffer from labeling bias.

4.2 Baselines

We compare NEEDLE with the following baselines (All pre-trained models used in the baseline methods have been continually pre-trained on the in-domain unlabeled data (i.e., Stage I of NEEDLE) for fair comparison):

Supervised Learning Baseline: We directly ne-tune the pre-trained model on the strongly labeled data. For E-commerce query NER, we use Query-RoBERTa-CRF, which is adapted from the RoBERTa large model. For E-commerce multilingual query NER, we use Query-mBERT-CRF, which is adapted from the mBERT. For Biomedical NER, we use BioBERT-CRF (Lee et al., 2020), which is adapted from BERT-base.

Semi-supervised Self-Training (SST): SST use the model obtained by supervised learning to generate pseudo labels for the unlabeled data and then conduct semi-supervised leaning (Wang et al., 2020; Du et al., 2021).

Weakly Supervised Learning (WSL): Simply combining strongly labeled data with weakly labeled data (Mann and McCallum, 2010).

Weighted WSL: WSL with weighted loss, where weakly labeled samples have a xed different weight :

$$\frac{\sum_{m}^{M} \ell(Y_m, f(X_m; \;)) + \sum_{m}^{\widetilde{M}} \ell(\widetilde{Y}^{w}_m, f(\widetilde{X}_m; \;))}{M + \widetilde{M}}.$$

We tune the weight and present the best result.

Robust WSL: WSL with mean squared error loss function, which is robust to label noise (Ghosh et al., 2017). As the robust loss is not compatible with CRF, we use the token-wise classi cation model for the Stage II training.

Partial WSL: WSL with non-entity weak labels excluded from the training loss (Shang et al., 2018).

4.3 E-commerce NER

We use span-level precision/recall/F1-score as the evaluation metrics. We present the main results on English query NER in Table 2.

Method	P	R	F1
NEEDLE	80.71	80.55	80.63
Supervised Baseline			
Query-RoBERTa-CRF	79.27	79.24	79.25
Semi-supervised Baseline			
SST	79.61	79.37	79.75
Weakly Supervised Baselines			
WSL	73.95	50.20	59.81
Weighted WSL[y]	78.07	64.41	70.59
Partial WSL	71.95	68.56	70.21
Weighted Partial WSL[y]	76.28	76.34	76.31
Robust WSL	66.71	42.78	52.13

Table 2: Main Results on E-commerce English Query NER: Span-level Precision/Recall/F1. [y]: we presented the results of the best weight, see results for all weights in Appendix B.

4.3.1 Main Results

NEEDLE: NEEDLE outperforms the fully supervised baseline and achieves the best performance among all baseline methods;

Weakly Supervised Baselines: All weakly supervised baseline methods, including WSL, Weighted WSL, Partial WSL and Robust WSL, lead to worse performance than the supervised baseline. This is consistent with our claim in Section 1. The weakly labeled data can hurt the model performance if they are not properly handled;

SST: Semi-supervised self-training outperforms the supervised baseline and weakly supervised baselines. This indicates that if not properly handled, the weak labels are even worse than the pseudo label generated by model prediction. In contrast, NEEDLE outperforms SST, which indicates that the weak labels can indeed provide additional knowledge and improve the model performance when their noise can be suppressed.

4.3.2 Ablation

We study the effectiveness of each component of NEEDLE. Speci cally, we use the following abbreviation to denote each component of NEEDLE:

WLC: Weak label completion.

NAL: Noise-aware loss function, i.e., Eq.(4). Since NAL is built on top of WLC, the two components need to be used together.

FT: Final ne-tuning on strongly labeled data (Stage III).

As can be seen from Table 3, all components are effective, and they are complementary to each other.

Method	P	R	F1
NEEDLE w/o FT/WLC/NAL	73.95	50.20	59.81
NEEDLE w/o FT/NAL	75.53	76.45	75.99
NEEDLE w/o FT	75.86	76.56	76.21
NEEDLE w/o WLC/NAL	80.03	79.72	79.87
NEEDLE w/o NAL	80.07	80.36	80.21
NEEDLE	80.71	80.55	80.63

Table 3: Ablation Study on E-commerce English Query NER.

4.3.3 Extension to Multilingual NER

The proposed framework can be naturally extended to improve multilingual NER. See details about the algorithm in Appendix D. The results of E-commerce Multilingual NER is presented in Table 4. As can be seen, the proposed NEEDLE outperforms other baseline methods in all 5 languages.

Method	En	Fr	It	De	Es
NEEDLE	78.17	75.98	79.68	78.83	79.49
w/o NAL	78.00	76.02	79.19	78.58	79.23
w/o WLC/NAL	77.68	75.31	78.22	77.99	78.22
w/o FT	73.88	72.96	75.44	76.51	76.87
w/o FT/NAL	73.87	72.56	75.26	76.11	76.62
Supervised Baseline					
Query-mBERT-CRF	77.19	74.82	78.11	77.77	78.11
Semi-supervised Baseline					
SST	77.42	75.21	77.82	78.10	78.65
Weakly supervised Baseline					
WSL	58.35	59.90	60.98	61.66	63.14

Table 4: E-commerce Multilingual Query NER: Span Level F1. See other metrics in Appendix D.

4.4 Biomedical NER

We present the main results on Biomedical NER in Table 5. NEEDLE achieves the best performance among all comparison methods. We outperform previous SOTA (Lee et al., 2020; Gu et al., 2020) by 0.41%, 5.07%, 3.15%, on BC5CDR-chemical, BC5CDR-disease and NCBI-disease respectively, in terms of the F1-score. We achieve very signi cant improvement on BC5CDR-disease. We conjecture that the weak labels for disease entities are relatively accurate, since WSL can also improve the model performance.

4.5 Analysis

Size of Weakly Labeled Data To demonstrate that NEEDLE can better exploit the weakly labeled

Method	BC5CDR chemical	BC5CDR disease	NCBI disease
NEEDLE	93.74	90.69	92.28
w/o NAL	93.60	90.07	92.11
w/o WLC/NAL	93.08	89.83	91.73
w/o FT	82.03	87.86	89.14
w/o FT/NAL	81.75	87.85	88.86
Supervised Baseline			
BioBERT-CRF	92.96	85.23	89.22
Semi-supervised Baseline			
SST	93.06	85.56	89.42
Weakly-supervised Baseline			
WSL	85.41	88.96	78.84
Reported F1-scores in Gu et al. (2020).			
BERT	89.99	79.92	85.87
BioBERT	92.85	84.70	89.13
SciBERT	92.51	84.70	88.25
PubMedBERT	93.33	85.62	87.82
Reported F1-scores in Nooralahzadeh et al. (2019).			
NER-PA-RL^{y}	89.93		-

Table 5: Main Results on Biomedical NER: Span Level F1-score. We also provide previous SOTA performance reported in Gu et al. (2020) and Nooralahzadeh et al. (2019).. ^{y}: NER-PA-RL is a WSL variant using instance selection. Nooralahzadeh et al. (2019) only report the averaged F1 of BC5CDR-chemical and BC5CDR-disease. See other metrics in Appendix C.

data, we test the model performance with randomly sub-sampled weakly labeled data. We plot the F1-score curve for E-commerce English query NER in Figure 2a and BC5CDR data in Figure 2b. We nd that NEEDLE gains more bene ts from increasing the size of weakly labeled data compared with other methods (SST and WSL). We also present the performance of NEEDLE w/o FT in Figure 2c. As can be seen, although the performance of NEEDLE w/o FT decreases with more weakly labeled data, the model can still learn more useful information and achieves better performance after ne-tuning.

Two Rounds of Stage II Training. Since the model after the nal ne-tuning is better than the initial model in Stage II, we study whether using the ne-tuned model for an addition round of Stage II can further improve the performance of NEEDLE. Speci cally, after Stage III, we 1) use the new model to complete the original weak labels; 2) conduct noise-aware continual pre-training over both strongly and weakly labeled data; 3) ne-tune the model on strongly labeled data. The results are presented in Figure 2 (last point of each curve). As can be seen, NEEDLE can obtain slight improvement using the two rounds of Stage II training. On the other hand, we also show that SST and NEEDLE w/o NAL achieve little improvement using the second round of training.

Size of Strongly Labeled Data To demonstrate that NEEDLE is sample ef cient, we test NEEDLE on randomly sub-sampled strongly labeled data on E-commerce NER. As we show in Figure 3, NEEDLE only requires 30% 50% strongly labeled data to achieve the same performance as the (fully) supervised baseline. We also observe that NEEDLE achieves more signi cant improvement with fewer labeled data: +2.28/3.64 F1-score with 1%/10% labeled data.

4.6 Weak Label Errors in E-commerce NER

Here we study several possible errors of the weak labels to better understand the weak labels and how the proposed techniques reduce these errors.

Label Distribution Mismatch . First, we show the distribution difference between the weak labels and the strong labels, and demonstrate how the weak label completion reduces the gap. Speci cally, we compare the entity distribution of the true labels, weak labels, corrected weak labels and self-training pseudo labels in Figure 4. As can be seen, the original weak labels suffer from severe missing entity issue (i.e., too many non-entity labels) and distribution shift (e.g., nearly no Misc labels). On the other hand, the corrected weak labels suffer less from the missing entities and distribution shift. SST pseudo labels are the most similar to the strong labels, which explains why SST can directly improves the performance.

Systematical Errors. We observe that many errors from the weakly labeled data are systematical errors, which can be easily xed by the nal ne-tuning stage. For example, "amiibo" is one Product Line of "nintendo". The amiibo characters should be de ned as Misc type, while the weak labels are all wrongly annotated as Color . We list 4 queries and their strong labels and weak labels in Table 6. Although these errors lead to worse performance in Stage II, they can be easily xed in the nal ne-tuning stage. Speci cally, the pre-training rst encourages the model to learn that "xxx amiibo" is a combination of color + productLine with a large amount of weakly labeled data, and then the ne-tuning step corrects such a pattern to misc + productLine with a limited amount of data. It is easier than directly learning the misc + productLine with the limited strongly labeled data.

(a) E-commerce En Query NER

(b) BC5CDR-disease

(c) w/ vs. w/o nal ne-tuning

Figure 2: Size of weakly labeled data vs. Performance. We present the performance after the nal round of ne-tuning in (a) and (b). We also compare the performance with and without ne-tuning in (c) using E-commerce English query NER data. The baselines are Query-RoBERTa-CRF for (a,c) and BioBERT-CRF for (b). “Baseline”: the baseline here is the fully supervised baseline. We also present the performance after two rounds of Stage II training at the rightmost point of each curve (“Stage II x2”).

Label Types	Querys and Labels			
Human Labels	zelda amiibo	wario amiibo	yarn yoshi amiibo	amiibo donkey kong
Original Weak Labels	zelda amiibo	wario amiibo	yarn yoshi amiibo	amiibo donkey kong
Corrected Weak Labels	zelda amiibo	wario amiibo	yarn yoshi amiibo	amiibo donkey kong
Self-Training Labels	zelda amiibo	wario amiibo	yarn yoshi amiibo	amiibo donkey kong

Table 6: Query Examples of “amiibo”. Entity Labels: Red: Misc, Blue: Product Line, Green: Color, Black: Non Entity, Orange: Media Title.

Figure 3: Performance vs. Size of Strongly Labeled Data. See detailed numbers in Appendix B.

Figure 4: Entity Distribution

Entity BIO Sequence Mismatch in Weak Label Completion. Another error of the weakly labels is the mismatched entity BIO sequence in the weak label completion step, e.g., B-productType followed by I-color [4]. For English Query NER, the proportion of these broken queries is 1.39%. Removing these samples makes the Stage II perform better (F1 score +1.07), while it does not improve the nal stage performance (F1 score -0.18). This experiment indicates that the nal ne-tuning suf ces to correct these errors, and we do not need to strongly exclude these samples from Stage II.

Quantify the Impact of Weak Labels. Here we examine the impact of weak labels via the lens of prediction error. We check the errors made by the model on the validation set. There are 2384 entities are wrongly classi ed by the initial NER model. After conducting NEEDLE, 454 of 2384 entities are correctly classi ed. On the other hand, the model makes 311 more wrong predictions. Notice that not all of them are directly affected by the weakly labeled data, i.e., some entities are not ob-

[4] E.g., Original Weak Labels: B-productType, O, O; Model Prediction: B-color, I-color, O; Corrected Weak Labels: B-productType, I-color, O.

served in the weakly labeled data. Some changes may be only due to the data randomness. If we exclude the entities which are not observed in the weakly annotated entities, there are 171 new correctly classi ed entities and 93 new wrongly classi- ed entities, which are affected by the weak labels. Such a ratio 171=93 = 1:84 >> 1 justi es that the advantage of NAL signi cantly out-weights the disadvantage of the noise of weak labels.

5 Discussion and Conclusion

Our work is closely related to fully weakly supervised NER. Most of the previous works only focus on weak supervision without strongly labeled data (Shang et al., 2018; Lan et al., 2020a; Liang et al., 2020). However, the gap between a fully weakly supervised model and a fully supervised model is usually huge. For example, a fully supervised model can outperform a weakly supervised model (AutoNER, Shang et al. (2018)) with only 300 articles. Such a huge gap makes fully weakly supervised NER not practical in real-world applications.

Our work is also relevant to semi-supervised learning, where the training data is only partially labeled. There have been many semi-supervised learning methods, including the popular self-training methods used in our experiments for comparison (Yarowsky, 1995; Rosenberg et al., 2005; Tarvainen and Valpola, 2017; Miyato et al., 2018; Meng et al., 2018; Clark et al., 2018; Yu et al., 2021). Different from weak supervision, these semi-supervised learning methods usually has a partial set of labeled data. They rely on the labeled data to train a suf ciently accurate model. The unlabeled data are usually used for inducing certain regularization to further improve the generalization performance. Existing semi-supervised learning methods such as self-training doesn't leverage the knowledge from weak supervision and can only marginally improve the performance.

Different from previous studies on fully weakly supervised NER, we identify an important research question on weak supervision: the weakly labeled data, when simply combined with the strongly labeled data during training, can degrade the model performance. To address this issue, we propose a new computational framework named NEEDLE, which effectively suppresses the extensive noise in the weak labeled data, and learns from both strongly labeled data and weakly labeled data. Our proposed framework bridges the supervised NER and weakly supervised NER, and harnesses the power of weak supervision in a principled manner. Note that, NEEDLE is complementary to fully weakly supervised / semi-supervised learning. One potential future direction is to combine NEEDLE with other fully weakly supervised / semi-supervised learning techniques to further improve the performance, e.g., contrastive regularization (Yu et al., 2021).

Broader Impact

This paper studies NER with small strongly labeled and large weakly labeled data. Our investigation neither introduces any social/ethical bias to the model nor ampli es any bias in the data. We do not foresee any direct social consequences or ethical issues.

References

B Carpenter. 2009. Coding chunkers as taggers: Io, bio, bmewo, and bmewo+. LingPipe Blog, page 14.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1914–1925, Brussels, Belgium. Association for Computational Linguistics.

Gamal Crichton, Sampo Pyysalo, Billy Chiu, and Anna Korhonen. 2017. A neural network multi-task learning approach to biomedical named entity recognition. BMC bioinformatics, 18(1):368.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: a resource for disease name recognition and concept normalization. Journal of biomedical informatics, 47:1–10.

Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Veselin Stoyanov, and Alexis Conneau. 2021. Self-training improves pre-training for natural language understanding. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 5408–5418, Online. Association for Computational Linguistics.

Jason Fries, Sen Wu, Alex Ratner, and Christopher Ré. 2017. Swellshark: A generative model for biomedical named entity recognition without labeled data. *arXiv preprint arXiv:1704.06360*.

Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. 2017. Robust loss functions under label noise for deep neural networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 1919–1925. AAAI Press.

Athanasios Giannakopoulos, Claudiu Musat, Andreea Hossmann, and Michael Baeriswyl. 2017. Unsupervised aspect term extraction with B-LSTM & CRF using automatically labelled datasets. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 180–188, Copenhagen, Denmark. Association for Computational Linguistics.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing. *arXiv preprint arXiv:2007.15779*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann.

Ouyu Lan, Xiao Huang, Bill Yuchen Lin, He Jiang, Liyuan Liu, and Xiang Ren. 2020a. Learning to contextually aggregate multi-source supervision for sequence labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2134–2146, Online. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020b. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Robert Leaman and Graciela Gonzalez. 2008. Banner: an executable survey of advances in biomedical named entity recognition. In *Biocomputing 2008*, pages 652–663. World Scientific.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Qi Li, Haibo Li, Heng Ji, Wen Wang, Jing Zheng, and Fei Huang. 2012. Joint bilingual name tagging for parallel corpora. In *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 1727–1731. ACM.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. BOND: bert-assisted open-domain named entity recognition with distant supervision. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1054–1064. ACM.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany. Association for Computational Linguistics.

Gideon S Mann and Andrew McCallum. 2010. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *Journal of machine learning research*, 11(2).

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-supervised neural text classification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 983–992. ACM.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE T-PAMI*, 41(8):1979–1993.

Farhad Nooralahzadeh, Jan Tore Lønning, and Lilja Øvrelid. 2019. Reinforcement-based denoising of distantly supervised NER with partial annotation. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 225–233, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683.

Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. 2005. Semi-supervised self-training of object detection models. In WACV, pages 29–36.

Jingbo Shang, Liyuan Liu, Xiaotao Gu, Xiang Ren, Teng Ren, and Jiawei Han. 2018. Learning named entity tagger using domain-specific dictionary. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2054–2064, Brussels, Belgium. Association for Computational Linguistics.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pages 1195–1204.

Yaqing Wang, Subhabrata Mukherjee, Haoda Chu, Yuancheng Tu, Ming Wu, Jing Gao, and Ahmed Hassan Awadallah. 2020. Adaptive self-training for few-shot neural sequence labeling. arXiv preprint arXiv:2010.03680.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2015. Overview of the biocreative v chemical disease relation (cdr) task. In Proceedings of the fifth BioCreative challenge evaluation workshop, volume 14.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. ArXiv, pages arXiv–1910.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In 33rd Annual Meeting of the Association for Computational Linguistics, pages 189–196, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. 2021. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1063–1077, Online. Association for Computational Linguistics.

Bianca Zadrozny and Charles Elkan. 2001. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001, pages 609–616. Morgan Kaufmann.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, pages 19–27. IEEE Computer Society.

A Estimation of Weak Label Confidence

Here we describe how do we estimate the confidence of weak labels — $\hat{P}(\mathbf{Y}^c = \mathbf{Y}|\mathbf{X})$. Notice that, the corrected weak labels $\mathbf{Y}^c$ in NEEDLE consists of two parts: original weak labels $\mathbf{Y}^w$ and model prediction $\mathbf{Y}^p$. So we estimate the confidence of corrected weak labels by the confidence of these two parts using a simple linear combination:

$$\hat{P}(\mathbf{Y}^c = \mathbf{Y}|\mathbf{X}) = \frac{\#\{\text{Matched Tokens}\}}{\#\{\text{Total Tokens}\}}\hat{P}(\mathbf{Y}^w = \mathbf{Y}|\mathbf{X}) + (1 - \frac{\#\{\text{Matched Tokens}\}}{\#\{\text{Total Tokens}\}})\hat{P}(\mathbf{Y}^p = \mathbf{Y}|\mathbf{X})$$

The weight of such linear combination comes from the rule of the weak label completion procedure. Recall that, we use the original weak labels for all matched tokens in original weakly-supervised data, while we use the model prediction for other tokens.

We first assume the confidence of weak labels are high, i.e. $\hat{P}(\mathbf{Y}^w = \mathbf{Y}|\mathbf{X}) = 1$, as there is less ambiguity in the domain-specific dictionary and matching process.

The label prediction $\mathbf{Y}^p$ of CRF model is based on Viterbi decoding score

$$\mathbf{Y}^p = \arg\max_{\mathbf{Y}} s(\mathbf{Y}) = \text{Decode}(\mathbf{Y}; f(\mathbf{X}; \ ))$$

The confidence of $\mathbf{Y}^p$, i.e., $\hat{P}(\mathbf{Y}^p = \mathbf{Y}|\mathbf{X})$ can be estimated via histogram binning (Zadrozny and Elkan, 2001). Specifically, we categorize samples into bins based on the decoding score $s(\mathbf{Y}^p)$. For each bin we estimate the confidence using a validation set (independent of the final evaluation set). For a new sample, we first calculate the decoding score, and estimate the prediction confidence by the confidence of the corresponding bin in the histogram. Figure 5 illustrates an example of histogram binning. As can be seen, the decoding score has a strong correlation with the prediction confidence.

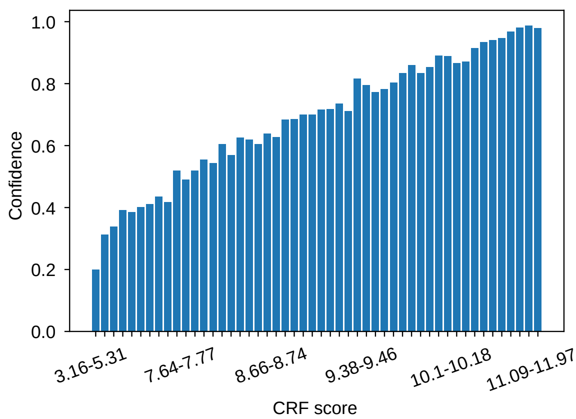

Figure 5: Decoding Score vs. Accuracy/Confidence

Finally, we enforce a smoothing when estimating the confidence. Specifically, we always make a conservative estimation by a post-processing:

$$P(\mathbf{Y}^c = \mathbf{Y}|\mathbf{X}) = \min(0.95, P(\mathbf{Y}^c = \mathbf{Y}|\mathbf{X}))$$

We enforce such a smoothing to count any potential errors (e.g., inaccurate original weak labels) and prevent model from overfitting. The smoothing parameter is fixed as 0.95 throughout the experiments.

B Additional Experimental Results for E-commerce NER

We also present Token/Span/Query level Accuracy, as they are commonly used in E-commerce NER tasks.

Method	Span P/R/F1	T/S/Q Accu.
RoBERTa (Supervised Baseline)	78.51/78.54/78.54	85.51/79.14/66.90
Weighted WSL		
weight = 0.5	75.38/52.94/62.20	61.07/52.61/37.32
weight = 0.1	77.31/57.85/66.18	65.65/57.70/43.83
weight = 0.01	78.07/64.41/70.59	71.75/64.43/52.52
Weighted Partial WSL		
weight = 0.5	72.94/71.77/72.35	81.10/72.53/59.14
weight = 0.1	75.24/74.68/74.96	83.08/75.36/62.50
weight = 0.01	76.28/76.34/76.31	84.14/76.94/63.91

Table 7: Performance of BERT (Supervised Baseline), Weighted WSL & Weighted Partial WSL on E-commerce English Query NER

B.1 Performance vs. Strongly Labeled Data

Method	Span P/R/F1	T/S/Q Accu.
(1%) Query-RoBERTa-CRF (30 epochs)	68.69/70.59/69.63	79.03/71.25/54.36
(10%) Query-RoBERTa-CRF (3 epochs)	71.69/73.72/72.69	81.90/74.26/58.36
(20%) Query-RoBERTa-CRF (3 epochs)	75.16/75.90/75.53	83.65/76.43/62.42
(50%) Query-RoBERTa-CRF (3 epochs)	76.95/77.90/77.42	84.88/78.41/64.96
(1%) NEEDLE	71.20/72.64/71.91	80.74/73.26/57.40
(10%) NEEDLE	76.25/76.15/76.20	84.09/76.67/63.79
(20%) NEEDLE	77.93/77.75/77.84	85.06/78.28/65.88
(50%) NEEDLE	79.12/79.23/79.18	85.92/79.73/67.77

Table 8: Performance vs. Size of Strongly Labeled Data on E-commerce English Query NER

C Additional Experimental Results for Biomedical NER

Method	BC5CDR-chem	BC5CDR-disease	NCBI-disease
Reported F1-scores of Baselines (Gu et al., 2020). Previous SOTA: PubMedBERT/BioBERT.			
BERT	-/-/89.99	-/-/79.92	-/-/85.87
BioBERT	-/-/92.85	-/-/84.70	-/-/89.13
SciBERT	-/-/92.51	-/-/84.70	-/-/88.25
PubMedBERT	-/-/93.33	-/-/85.62	-/-/87.82
Re-implemented Baselines			
BERT	88.55/90.49/89.51	77.54/81.87/79.64	83.50/88.54/85.94
BERT-CRF	88.59/91.44/89.99	78.70/81.53/80.09	85.33/86.67/85.99
BioBERT	92.59/93.11/92.85	82.36/86.66/84.45	86.75/90.83/88.74
BioBERT-CRF	92.64/93.28/92.96	83.73/86.80/85.23	87.18/91.35/89.22
Based on BioBERT and CRF layer			
SST	92.40/93.74/93.06	84.01/87.18/85.56	87.00/91.98/89.42
WSL	82.17/88.91/85.41	90.72/87.27/88.96	87.14/71.98/78.84
NEEDLE w/o WLC/NAL	**92.85**/93.31/93.08	91.37/88.34/89.83	**91.68**/91.77/91.73
NEEDLE w/o FT/NAL	79.29/84.38/81.75	82.44/**94.03**/87.85	87.17/90.62/88.86
NEEDLE w/o NAL	**92.93**/94.28/**93.60**	86.73/93.69/90.07	**91.82**/92.40/**92.11**
NEEDLE w/o FT	79.87/84.31/82.03	82.39/**94.12**/87.86	87.31/91.04/89.14
NEEDLE	**92.89/94.60/93.74**	**87.99**/93.56/**90.69**	**91.76/92.81/92.28**

Table 9: Main Results on Biomedical NER: Span Precision/Recall/F1. The *Best* performance is **bold**, and the results that are close to best performance ($\leq 0.2\%$) are also **bold**.

C.1 Additional Baseline Results

We compare NEEDLE with other popular semi-supervised (Mean-Teacher, Tarvainen and Valpola (2017), and VAT, Miyato et al. (2018)) and weakly supervised baselines (BOND, Liang et al. (2020)) [5].

Method	NEEDLE	Mean-Teacher	VAT	BOND	BOND + FT (Stage III)
F1-score	93.74	92.88	93.10	86.93	92.82

Table 10: F1-score on BC5CDR-chem.

[5] https://github.com/cliang1453/BOND/pull/12

D Extension: Multilingual NER

The proposed framework can be extended to improve multilingual NER. For Stage I and Stage II, we use data from other languages to learn domain-specific knowledge and task-related knowledge. In the final fine-tuning stage, we use the data from the target language, which allows us to adapt the model to the target language and obtain a better performance on the target language. The framework is summarized in Figure 6. The results of Multilingual Query NER are presented in Table 11. As can be seen, NEEDLE outperforms baseline methods.

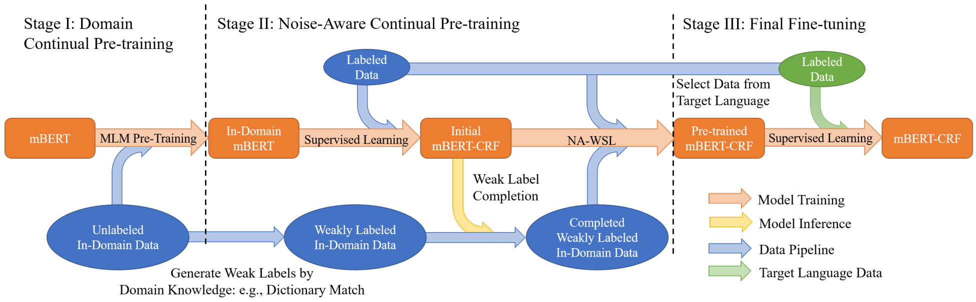

Figure 6: Three-Stage NEEDLE for Multilingual NER

Method (*Span P/R/F1*)	En	Fr	It	De	Es
mBERT-CRF (Single)	76.14/76.04/76.09	72.87/73.00/72.93	76.95/77.67/77.31	74.74/78.08/76.37	76.34/76.75/76.54
mBERT-CRF	76.38/76.25/76.31	74.69/75.06/74.87	77.82/77.60/77.71	75.93/78.52/77.20	78.18/77.57/77.87
Query-mBERT-CRF	77.21/77.18/77.19	74.59/75.05/74.82	78.22/78.01/78.11	76.46/79.12/77.77	78.50/77.73/78.11
Based on Query-mBERT and CRF layer					
SST	77.52/77.33/77.42	75.15/75.28/75.21	78.00/77.64/77.82	76.82/79.43/78.10	79.14/78.17/78.65
WSL	74.20/48.09/58.35	71.17/51.71/59.90	74.72/51.51/60.98	74.34/52.68/61.66	76.32/53.85/63.14
NEEDLE w/o WLC/NAL	77.89/77.47/77.68	75.28/75.35/75.31	78.17/78.28/78.22	76.68/79.33/77.99	78.29/78.14/78.22
NEEDLE w/o FT/NAL	72.73/75.06/73.87	72.00/73.12/72.56	75.19/75.34/75.26	74.65/77.63/76.11	77.07/76.18/76.62
NEEDLE w/o NAL	**78.27/77.74/78.00**	**76.09/75.95/76.02**	79.14/79.25/79.19	77.55/79.63/78.58	79.60/78.86/79.23
NEEDLE w/o FT	72.79/75.01/73.88	72.46/73.46/72.96	75.39/75.50/75.44	75.09/77.98/76.51	77.46/76.29/76.87
NEEDLE	**78.40/77.95/78.17**	**76.05/75.91/75.98**	**79.61/79.76/79.68**	**77.79/79.90/78.83**	**79.85/79.13/79.49**
Method (*T/S/Q Accu.*)	En	Fr	It	De	Es
mBERT-CRF (Single)	83.26/76.80/61.68	80.27/72.91/57.48	83.70/78.13/60.75	79.53/76.38/60.72	83.58/77.56/59.64
mBERT-CRF	83.37/76.97/62.21	81.43/74.92/60.35	84.31/78.06/60.65	80.48/76.82/62.47	84.94/78.23/61.44
Query-mBERT-CRF	84.15/77.85/63.44	81.36/74.91/60.17	84.83/78.46/61.26	80.93/77.40/62.81	85.20/78.27/62.12
Based on Query-mBERT and CRF layer					
SST	84.18/78.02/63.57	81.66/75.12/60.92	84.45/78.13/60.89	81.26/77.72/63.61	85.35/78.56/62.90
WSL	54.40/47.43/28.97	59.11/51.08/32.85	59.79/50.59/30.75	56.16/51.16/33.59	61.36/53.29/32.48
NEEDLE w/o WLC/NAL	84.42/78.12/64.43	81.65/75.24/60.74	84.76/78.65/61.77	81.32/77.59/63.37	84.82/78.84/61.95
NEEDLE w/o NAL/FT	83.46/75.80/57.93	81.20/73.04/56.90	83.48/75.97/57.22	80.31/76.00/60.79	83.90/76.80/59.30
NEEDLE w/o NAL	**84.63/78.42/64.76**	**82.34/75.83/61.91**	85.34/79.63/63.17	81.68/77.90/64.34	85.64/79.48/63.41
NEEDLE w/o FT	83.50/75.76/58.01	80.92/73.38/57.34	83.45/76.03/57.39	80.48/76.31/61.22	84.10/76.97/60.12
NEEDLE	**84.74/78.59/64.86**	**82.14/75.80/61.96**	**85.65/80.12/63.71**	**81.79/78.15/64.84**	**86.00/79.80/64.03**

Table 11: E-commerce Multilingual Query NER: Span Precision/Recall/F1 and Token/Span/Query level Accuracy. The *Best* performance is **bold**, and the results that are close to best performance ($\leq 0.2\%$) are also **bold**. 'mBERT-CRF (Single)': fine-tune mBERT with strongly labeled data from the target language. 'w/ Fine-tune': the additional fine-tuning stage only use strongly labeled data from the target language. For other methods, we use multilingual human-annotated data.